**求和实验报告**

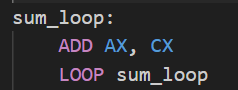
2251656 于思源

1. **将结果放在寄存器中求和1-100**
2. **基本思路**

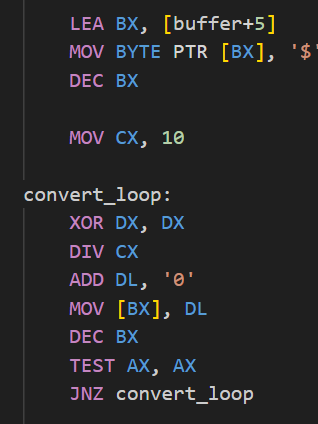
首先进行初始化，然后使用循环从1到100进行累加，并将结果存储在AX中，最后将累加结果转换为字符串形式并存入缓冲区 buffer。

1. **核心代码**

求和：



转化为字符串:



1. **结果**

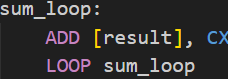


1. **将结果放在数据段中求和1-100**
2. **基本思路**

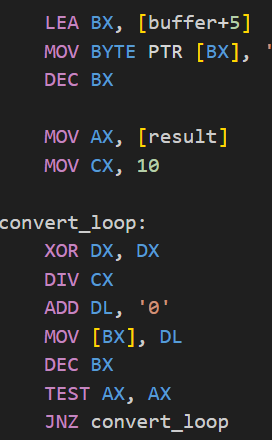
与寄存器中相同，区别只是将数据放于数据段中

1. **核心代码**

求和：



转化为字符串：



1. **结果**

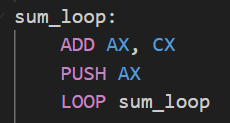


1. **将结果放在栈中求和1-100**
2. **基本思路**

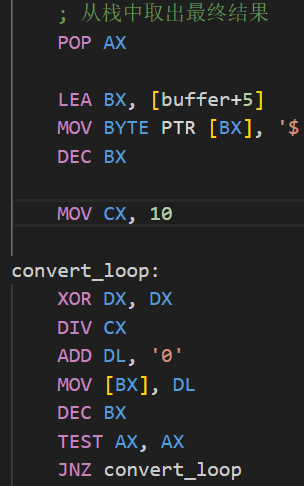
同上，区别只是将结果放在栈中。

1. **核心代码**

求和：



输出为字符串：



1. **结果**



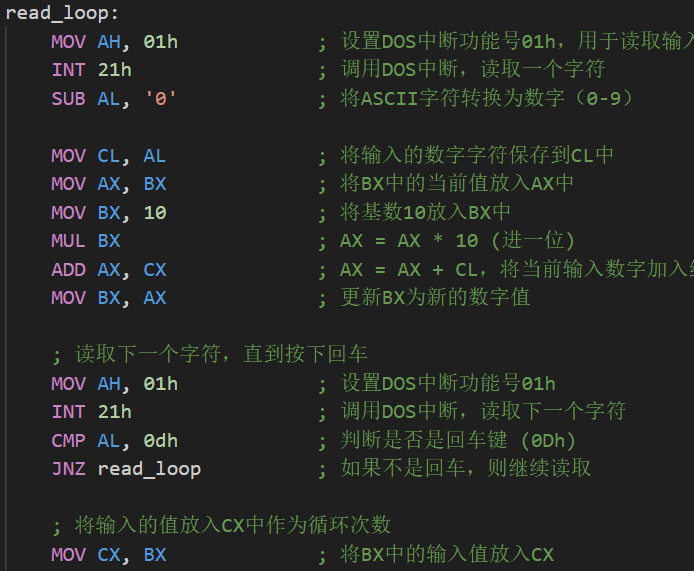
1. **用户输入1-100中数据num，求和1-num**
2. **基本思路**

其他思路同上，此处只给出读取num的思路。

逐个字符读取，并将ASCII转换为对应数字，并保存在CL寄存器中。最后将结果寄存器中的数据\*10，再加上CL寄存器中的值，持续进行循环，调用DOS终端，读取下一个字符，若字符不为回车，则继续读取。

1. **核心代码**

读取：



1. **结果**



1. **实验总结**

通过本次实验，我更深刻地掌握了循环指令和跳转指令的应用。